Proceedings of the IASTED International Conference
MODELLING AND SIMULATION
May 16-18, 2001, Pittsburgh, Pennsylvania, USA

Modelling Active Vision Systems for Dynamic Simulated Environments

Tamer F. Rabie

Department of Electrical and Computer Engineering
Ryerson Polytechnic University
350 Victoria Street
Toronto, Ontario, M5B 2K3, Canada
e-mail: tamer@ee.ryerson.ca

Demetri Terzopoulos

Department of Computer Science
University of Toronto
6 King's College Road
Toronto, Ontario, M5S 3H4, Canada
e-mail: dt@cs.toronto.edu

Abstract

This paper presents our work on modelling a vision system for highly mobile autonomous agents that is capable of dynamic obstacle avoidance and active perception. We demonstrate the robust performance of the system in artificial animals with directable, foveated eyes, situated in a physics-based simulated environment. Through active perception, each agent controls its eyes and body by continuously analyzing photorealistic binocular retinal image streams. The vision system estimates optical flow, computes stereo disparity and segments looming targets in the low-resolution visual periphery while controlling eye movements to track an object fixated in the high-resolution fovea. It matches segmented targets against mental models of colored objects of interest in order to decide whether the segmented objects are harmless or represent dangerous obstacles. The latter are localized, enabling the artificial animal to exercise the sensorimotor control necessary to support complex behaviors, such as predation, and obstacle avoidance.

Keywords: *Active Vision; Artificial Intelligence; Simulated Environment; Virtual Reality; Simulated Robotics; Artificial Animals.*

1 Introduction

Animals are active observers of their environment [14]. This fact has inspired a trend in computer vision popularly known as "active vision" [3, 4]. Our recently proposed *animat vision* paradigm offers a new approach to developing biologically inspired active vision systems and experimenting with them [29]. Rather than allow the limitations of available robot hardware to hamper research, animat vision prescribes the use of virtual robots that take the form of realistic artificial animals, or animats, situated in physics-based virtual worlds. Animats are autonomous virtual agents possessing highly mobile, muscle-actuated bodies and brains with motor, perception, behavior and learning centers. In the perception center of the animat's brain, computer vision algorithms continually analyze incoming perceptual information. Based on this analysis, the behavior center dispatches motor commands to the animat's body, thus forming a complete sensorimotor control system.

Motion and color play an important role in animal perception. Birds and insects exploit optical flow for obstacle avoidance and to control their ego-motion [14]. Some species of fish are able to recognize the color signatures of other fish and use this information in certain piscene behaviors [1]. The human visual system is highly sensitive to motion and color. We tend to focus our attention on moving colorful objects. Motionless objects whose colors blend in to the background are not as easily detectable, and several camouflage strategies in the animal kingdom rely on this fact [12].

Biological creatures move through the world with little apparent effort. Many do so using eyes with a high-acuity fovea covering only a small fraction of a visual field whose resolution decreases monotonically towards the periphery. Spatially nonuniform retinal imaging provides opportunities for increased computational efficiency through economization of photoreceptors and focus of attention, but it forces the visual system to solve problems that do not generally arise with a uniform field of view. A key problem is determining how to deal with objects that are detected in the low resolution periphery while focusing attention on an object of interest fixated in the high resolution fovea. In this paper we present a solution to this problem through the combined exploitation of color, motion and depth information from stereo disparity.

Building upon the animat vision paradigm, the stereo, motion and color based motor and gaze control algorithms that we propose in this paper are implemented and evaluated within artificial fishes in a virtual marine world. The fish animats are the result of research in the domain of artificial life (see [30] for the details). In the present work, the fish animat serves as an autonomous mobile robot inhabiting a photorealistic, dynamic environment. Our new navigation algorithms significantly enhance the prototype animat vision system implemented in prior work [28, 22, 29, 23]. They support in the artificial fishes more robust vision-guided navigation, including obstacle recognition and avoidance. We briefly review the animat vision system in the next section before presenting, in the subsequent sections, our work on integrating motion, stereo disparity and color analysis for animat navigation and perception.

2 A Prototype Animat Vision System

The basic functionality of the animat vision system, which is described in detail in [28, 29], starts with binocular perspective projection of the color 3D world onto the ani-

mat's 2D retinas. Retinal imaging is accomplished by photorealistic graphics rendering of the world from the animat's point of view. This projection respects occlusion relationships among objects. It forms spatially variant visual fields with high resolution foveas and progressively lower resolution peripheries. Based on an analysis of the incoming color retinal image stream, the visual center of the animat's brain supplies saccade control signals to its eyes to stabilize the visual fields during locomotion, to attend to interesting targets based on color, and to keep moving targets fixated. The artificial fish is thus able to approach and track other artificial fishes visually.

2.1 Eyes and Retinal Imaging

The artificial fish has binocular vision. The movements of each eye are controlled through two gaze angles (θ, ϕ) which specify the horizontal and vertical rotation of the eyeball, respectively. The angles are given with respect to the head coordinate frame, such that the eye is looking straight ahead when $\theta = \phi = 0°$.

Each eye is implemented as four coaxial virtual cameras to approximate the spatially nonuniform, foveal/peripheral imaging capabilities typical of biological eyes. Fig. 1(a) shows an example of the 64×64 images that are rendered by the coaxial cameras in each eye (rendering employs the OpenGL library and graphics pipeline on Silicon Graphics workstations). The level $l = 0$ camera has the widest field of view (about 120°) and the lowest resolution. The resolution increases and the field of view decreases with increasing l. The highest resolution image at level $l = 3$ is the fovea and the other images form the visual periphery. Fig. 1(b) shows the 512×512 binocular retinal images composited from the coaxial images at the top of the figure. To reveal the retinal image structure in the figure, we have placed a white border around each magnified component image. Vision algorithms which process the four 64×64 component images are 16 times more efficient than those that process a uniform 512×512 retinal image.

2.2 Foveation by Color Object Detection

The brain of the artificial fish stores a set of color models of objects that are of interest to it. For instance, if the fish is by habit a predator, it would possess models of prey fish. The mental models are stored as a list of 64×64 RGB color images.

To detect and localize any target that may be imaged in the low resolution periphery of its retinas, the animat vision system of the fish employs an improved version of a color indexing algorithm proposed by Swain [26].[1] Since each model object has a unique color histogram signature, it can be detected in the retinal image by histogram intersection and localized by histogram backprojection.

2.3 Saccadic Eye Movements

When a target is detected in the visual periphery, the eyes will saccade to the angular offset of the object to bring it within the fovea. With the object in the high resolution fovea, a more accurate foveation is obtained by a second pass of histogram backprojection. A second saccade typically centers the object accurately in both left and right foveas, thus achieving vergence. The saccades are performed by incrementing the gaze angles (θ, ϕ) in order to rotate the eyes to the required gaze direction.

2.4 Visual Field Stabilization using Optical Flow

It is necessary to stabilize the visual field of the artificial fish because its body undulates as it swims. Once a target is verged in both foveas, the stabilization process assumes the task of keeping the target foveated during locomotion.

Stabilization is achieved by computing the overall translational displacement (u, v) of intensities between the current foveal image and that from the previous time instant, and updating the gaze angles to compensate. The displacement is computed as a translational offset in the retinotopic coordinate system by a least squares minimization of the optical flow between image frames at times t and $t - 1$ [16].

The optical flow stabilization method is robust only for small displacements between frames. Consequently, when the displacement of the target between frames is large enough that the method is likely to produce bad estimates, the foveation module is invoked to re-detect and re-foveate the target as described earlier. Each eye is controlled independently during foveation and stabilization of a target. Hence, the two retinal images must be correlated to keep them verged accurately on the target.

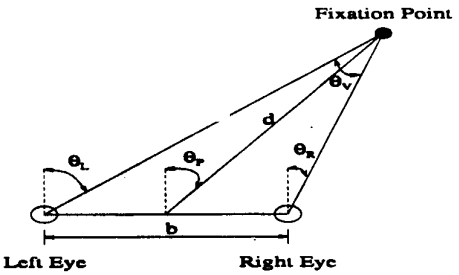


Figure 2: Gaze angles and range to target geometry.

2.5 Vision-Guided Navigation

The artificial fish can also employ the gaze direction (i.e., the gaze angles) while the eyes are fixated on a target to navigate towards the target. The θ angles are used to compute the left/right turn angle θ_P shown in Fig. 2, and the ϕ angles are similarly used to compute an up/down turn angle ϕ_P. The fish's turn motor controllers are invoked to execute a left/right turn—right-turn-MC for an above-threshold positive θ_P and left-turn-MC for negative θ_P—with $|\theta_P|$ as parameter. Up/down turn motor commands are issued to the fish's pectoral fins, with an above-threshold positive ϕ_P interpreted as "up" and negative as "down". The motor controllers are explained in [30].

The remainder of the paper presents our new work on integrating color, motion and disparity analysis within the animat vision system for dynamic perception and obstacle avoidance.

[1] Our improvements, which include iterative model histogram scaling and weighted histograms, make the technique much more robust against the large variations in scale that occur in our application. The details of the improved algorithm are presented in [29].

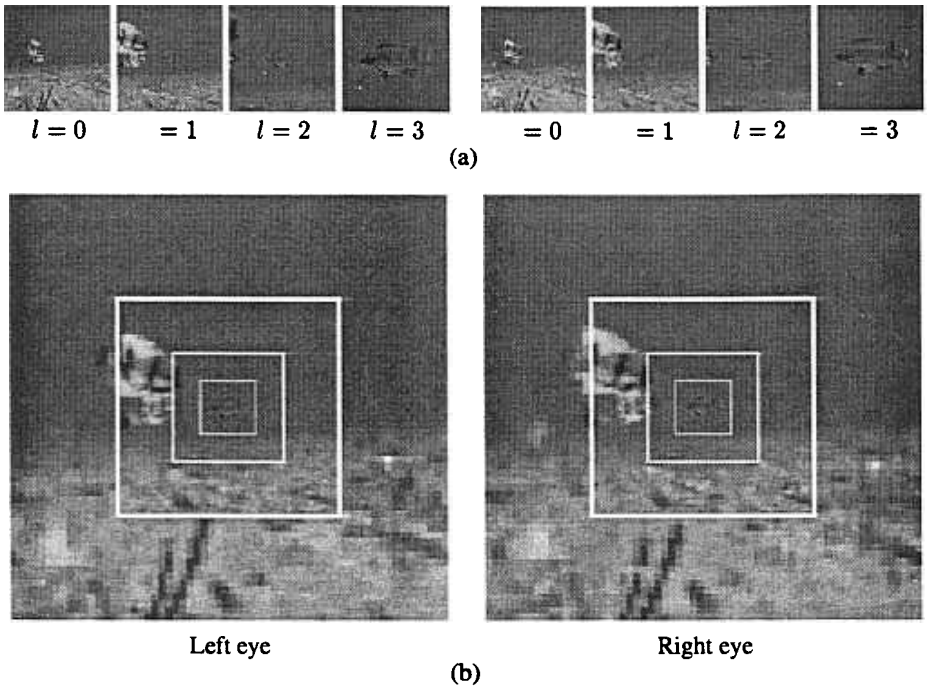


Figure 1: Binocular retinal imaging. (a) 4 component images; $l = 0, 1, 2$, are peripheral images; $= 3$ is foveal image. (b) Composited retinal images (borders of composited component images are shown in white).

3 Disparity and Color for Obstacle Avoidance

Color and stereo algorithms have been discussed extensively in the literature in a variety of passive vision systems, but rarely have they been integrated for use in dynamic obstacle avoidance systems. Color and stereo cues have recently been integrated together with motion cues to implement a real-time passive stereo system that can detect and identify moving objects for application to surveillance and human-computer interaction [2]. Disparity and color cues have also been combined to improve the focus of attention and recognition capabilities of an active vision system [15].

The following sections describe our dynamic obstacle recognition and avoidance algorithms. Exploiting stereo and color cues, the algorithms enable the animat to navigate through its virtual environment fixating and tracking a reference target in the fovea while avoiding obstacles that appear in its low resolution visual periphery.

3.1 Stereo Analysis

Stereo analysis is a process of extracting scene depth information by measuring the disparity of corresponding points between left and right binocular images. The task of determining the correspondence between points in the two views is known as the correspondence problem, which is considered difficult. Classical approaches to stereo analysis try to deal with the correspondence problem with two basic algorithms; area-based [20, 16] and feature-based approaches [21, 16]. Several approaches take into consideration available biological and neurophysiological data about the human visual system [21, 24, 19]. There is biological evidence that the pattern of light projected on the human retina is sampled and spatially filtered. Early in cortical visual processing, receptive fields become oriented and are well approximated by linear spatial filters, with impulse response functions that are similar to partial derivatives of a Gaussian function [32].

Our animat vision approach for estimating stereo disparity is motivated by knowledge about the early visual processing in the primate cortex. We implement the receptive fields as steerable spatial filters that process the input images. The steerable filter responses at an image location form a *feature vector* that is used in establishing correspondence. The outputs of a steerable filter convolved with an image at multiple orientations provides rich information about a local neighborhood around each pixel. This simplifies the matching of image patches from the left and right images of a stereo pair and the probability of a correct match increases as the length of the feature vector increases.

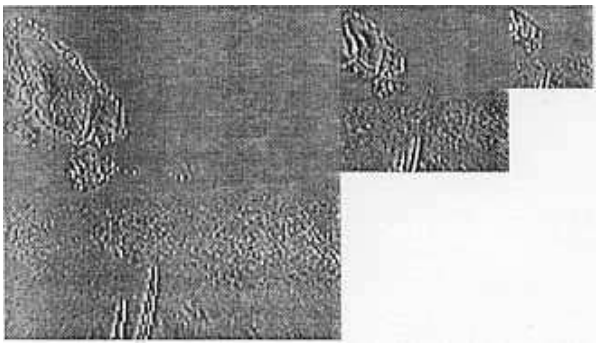
"Steerable filter" is a term used to describe a class of spatial filters in which a filter of arbitrary orientation is synthesized as a linear combination of a set of basis filters. Steerable filters, first developed by Freeman and Adelson [13], have recently been used for estimation of scene motion [17] and for object recognition [5] and stereopsis [19]. Simoncelli and Freeman have recently introduced a multi-scale, multi-orientation steerable filter image decomposition

framework called the Steerable Pyramid [25] which we use as a front-end for our stereo algorithm. It has the advantage of producing feature descriptions that are both translation- and rotation-invariant.

3.2 Disparity Estimation for Animat Vision

Our disparity estimation algorithm starts by decomposing the left and right images into steerable pyramid representations using the framework developed by Simoncelli and Freeman in [25]. Fig. 3 shows an example of a three-level steerable pyramid for a single orientation for an retinal image acquired by the animat's right eye.

(a)



(b)

Figure 3: (a) An image acquired by the animat's right eye, (b) A three-level Steerable Pyramid of the image in (a) shown for a single orientation.

Feature vectors $\mathbf{f}_R(x, y, l)$ and $\mathbf{f}_L(x, y, l)$ are then constructed from the right and left pyramid responses for each pixel at each level of the pyramid by combining the responses of the multi-orientation steerable filters at each pixel into a vector that provides a very rich description of the intensities at that pixel in the image. To further enrich the description of each pixel, we make use of the (R, G, B) color signals from our color images by including them in the feature vector. This simple addition improves our matching process considerably by restricting the matching process to areas of similar color composition, which can be considered as a sort of color-feature constraint.

An initial disparity map is estimated at each individual level by matching left and right feature vectors by minimizing the mean square error (MSE) between left and right feature vectors. The MSE measure is computed over all the elements in the vector as follows;

$$E_m = \frac{1}{S} \sum_{i \in S} [\mathbf{f}_R^i(x, y, l) - \mathbf{f}_L^i(x + d_x, y + d_y, l)]^2, \quad (1)$$

where S is the feature vector size. The MSE measure E_m is computed for a limited range of horizontal and vertical disparities $d_x(l) \in D_x(l)$ and $d_y(l) \in D_y(l)$.

A coarse-to-fine-flow-through strategy is then taken based on the assumption that for level l disparity estimates $|d(l)| > |\frac{D(l)}{4}|$ are more accurately estimated at the coarser level $l+1$. Thus at coarse levels, large disparities are estimated presumably more accurately, and these flow through to the finer levels, while small disparities that are estimated from the finer levels are assumed accurate since they cannot be estimated at coarser levels due to the loss of high frequency structure from the original coarse-level images. Fig. 4 shows a disparity map estimated by the algorithm for a stereo retinal image acquired by our animat.

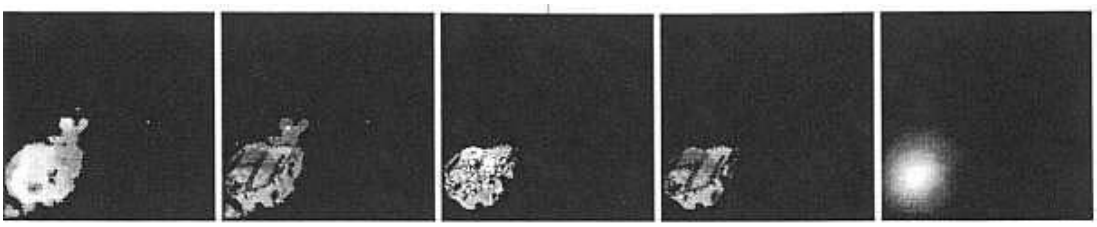
Figure 4: Right and Left images acquired by the animat's eye and the estimated disparity map

3.3 Color Obstacle Recognition and Localization

To distinguish between dangerous obstacles and benign objects we combine the disparity cues estimated using the above algorithm with color cues available naturally from the acquired photorealistic images. The animat continuously computes a disparity map from its stereo retinal input as it navigates through the virtual world. The estimated disparity map is used as a bottom-up cue to alert the animat of potential danger while color is used as a top-down cue for recognition of the actual obstacle.

The estimated disparity map is first segmented into potential obstacles via thresholding with an appropriate (empirically determined) threshold value. Then each segmented object is examined and matched against color mental models of designated dangerous obstacle objects. A match indicates that a candidate object is really an obstacle and is to be avoided, otherwise the object is considered harmless. Harmless objects include food particles and sea weeds.

The region of support of an obstacle's disparity map is then convolved with a circular disc of equal area. This will blur out any misclassified pixels in the segmentation while emphasizing the obstacle and facilitating its localization. The pixel location (x_c, y_c) of the peak in the blurred



(a) (b) (c) (d) (e)

Figure 5: (a) Thresholded disparity map of fig. 4, (b) Corresponding color segmentation of potential obstacles, (c) Backprojection map, (d) The exact region of support of the segmented obstacle, (e) The localization of the obstacle by blurring the corresponding segmentation of the disparity map.

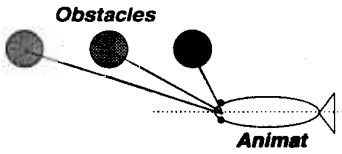


Figure 6: Relationship between close objects and large steering angles.

disparity map localizes the obstacle. Fig. 5 shows images of the various segmentation steps.

3.4 Obstacle Avoidance Strategy

The point of localization (x_c, y_c) obtained from the peak in the blurred disparity map is used to compute the steering angles the animat must use to steer clear of the obstacle. The angular location with respect to the right eye is given as $\theta = \tan^{-1}(\frac{x_c}{f_c}), \phi = \tan^{-1}(\frac{y_c}{f_c})$, where f_c is the camera focal length. The turn angles given to the animat's motor controller are, thus, proportional to $(-\theta, -\phi)$; i.e., in the opposite direction, to avoid collision while still fixating on a reference target to stabilize the visual world.

The merit of using $(-\theta, -\phi)$ for steering the animat is twofold: 1) simplicity of computing a steering vector and, 2) the fact that for close objects (θ, ϕ) is large, as is depicted in Fig. 6. Therefore, the turn maneuver will be large to avoid the obstacle quickly. The farther away the obstacle, the smaller the turn angles, hence steering will not be excessive.

Fig. 7 shows frames from a top view of a sequence showing the animat navigating in its environment. The animat is fixating and tracking a target red fish while avoiding obstacles taking the form of other fish obstructing its path. The figure shows two instances where the animat encounters an obstacle (frames 156 and 180). These are followed by frames showing how the animat has successfully avoided the obstacle by steering its body in the opposite direction as explained above. The white lines emanating from the eyes of the observer indicate the gaze direction.

Next, we present active vision additions to the animat vision system enhancing the animat's functionality in it's virtual environment. In the following section, motion and color cues are integrated to increase the robustness of the animat's perceptual functions. This, in addition to the stereo and color obstacle avoidance capabilities, further enhances the animat's behavioral responses.

4 Integrating Motion and Color for Attention

Selective attention is an important mechanism for dealing with the combinatorial aspects of search in vision. Deciding where to redirect the fovea can involve a complex search process [31]. In this section we offer an efficient solution which integrates motion and color to increase the robustness of our animat's perceptual functions.

Integrating motion and color for object recognition can improve the robustness of moving colored object recognition. Motion may be considered a bottom-up *alerting* cue, while color can be used as a top-down cue for model-based recognition [27]. Therefore, integrating motion and color can increase the robustness of the recognition problem by bridging the gap between bottom-up and top-down processes, thus, improving the selective attention of dynamic perceptual systems such as the animat vision system that we are developing.

4.1 Where to Look Next

Redirecting gaze when a target of interest appears in the periphery can be a complex problem. One solution would be to section the peripheral image into smaller patches or focal probes [10] and search of all the probes. The strategy will work well for sufficiently small images, but for dynamic vision systems that must process natural or photorealistic images the approach is not effective.

We choose a simple method based on motion cues to help narrow down the search for a suitable gaze direction [11]. We create a saliency image by initially computing a reduced optical flow field between two stabilized peripheral image frames (an advantage of the multiresolution retina is the small 64×64 peripheral image). Then an affine motion model is fitted to the optical flow using a robust regression method that is described in [22]. The affine motion parameters are fitted to the dominant background motion. A saliency map is determined by computing an error measure between the affine motion parameters and the estimated optical flow as follows:

$$S(x,y) = \sqrt{[u_a(x,y) - u(x,y)]^2 + [v_a(x,y) - v(x,y)]^2}, \quad (2)$$

Figure 7: An overhead view of a blue animat pursuing a red reference fish while detecting and avoiding other fish obstacles.

where (u, v) is the computed optical flow and

$$\begin{aligned} u_a(x,y) &= a + bx + cy, \\ v_a(x,y) &= d + ex + fy \end{aligned} \quad (3)$$

is the affine flow at retinal image position (x, y). The saliency image S is then convolved with a circular disk of area equal to the expected area of the model object of interest as it appears in the peripheral image.[2]

The blurring of the saliency image emphasizes the model object in the image. The maximum in S is taken as the location of the image probe. The image patches that serve as probes in consecutive peripheral frames form the image sequence that is processed by the motion segmentation module described later. Fig. 8 shows four consecutive peripheral images with the image probes outlined by white boxes. The blurred saliency image is shown at the end of the sequence in Fig. 8. Clearly the maximum (brightness) corresponds to the fast moving blue fish in the lower right portion of the peripheral image.

4.2 Robust Optical Flow

A key component of the selective attention algorithm is the use of optical flow. Given a sequence of time-varying images, points on the retina appear to move because of the relative motion between the eye and objects in the scene [14]. The vector field of this apparent motion is usually called optical flow [16]. Optical flow can provide important information about the spatial arrangement of objects viewed and the rate of change of this arrangement.

For our specific application, however, we require efficiency, robustness to outliers, and an optical flow estimate at all times. Recent work by Black and Anandan [7, 8] satisfies our requirements. They propose incremental minimization approaches using robust statistics for the estimation of optical flow which are geared towards dynamic environments. As is noted by Black, the goal is incrementally to integrate motion information from new images with previous optical flow estimates to obtain more accurate information about the motion in the scene over time. A detailed description of this method can be found in [6].

Ideally optical flow is computed continuously[3] as the animat navigates in its world, but to reduce computational cost

[2] Reasonably small areas suffice, since objects in the 64×64 peripheral image are typically small at peripheral resolution. Methods for estimating appropriate areas for the object, such as Jagersand's information theoretic approach [18], may be applicable.

[3] By continuously, we mean that there is an estimate of the optical flow at every time instant.

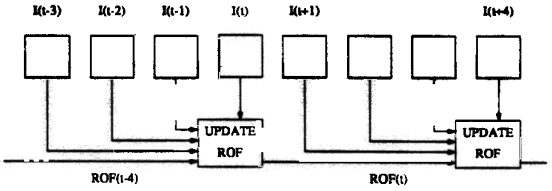


Figure 9: Incremental estimation of robust optical flow (ROF) over time.

we choose to update the current estimate of the optical flow every four frames. The algorithm is however still "continuous" because it computes the current estimate of the optical flow at time t using image frames at t-3, t-2, t-1, and t in a short-time batch process. Fig. 9 shows this more clearly. This arrangement requires storage of the previous three frames for use by the estimation module.

The flow at $t + 1$ is initialized with a predicted flow computed by forward warp of the flow estimate at t by itself[4] and then the optical flow at $t + 4$ is estimated by spatiotemporal regression over the four frames. Full details of our robust optical flow estimation algorithm can be found in [22].

4.3 Motion Segmentation and Color Recognition

For the animat to recognize objects moving in its periphery it must first detect their presence by means of a saliency map as described in section 4.1. Once it detects something that might be worth looking at, it must then segment its region of support out from the whole peripheral image and then match this segmentation with mental models of important objects. Fig. 10 shows the steps involved in an incremental segmentation of the detected object over the duration of the four probe images as explained above.

Segmentation of the optical flow at each time instant is performed by fitting an affine parametric motion model to the robust optical flow (ROF) estimated so far at the current time instant. This is done by incrementally minimizing the cost function given as

$$E(a,b,c,d,e,f) = E_x(a,b,c) + E_y(d,e,f), \quad (4)$$

where (a, b, c, d, e, f) are the affine motion parameters. E_x and E_y are formulated using robust estimation to account for

[4] The flow estimate is being used to warp itself, thus predicting what the motion will be in the future.

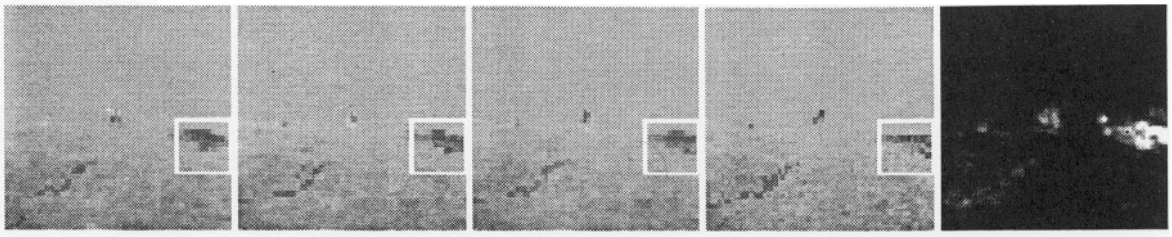


Figure 8: Four consecutive peripheral images with image probes outlined by white squares. Saliency image (right), with bright areas indicating large motions.

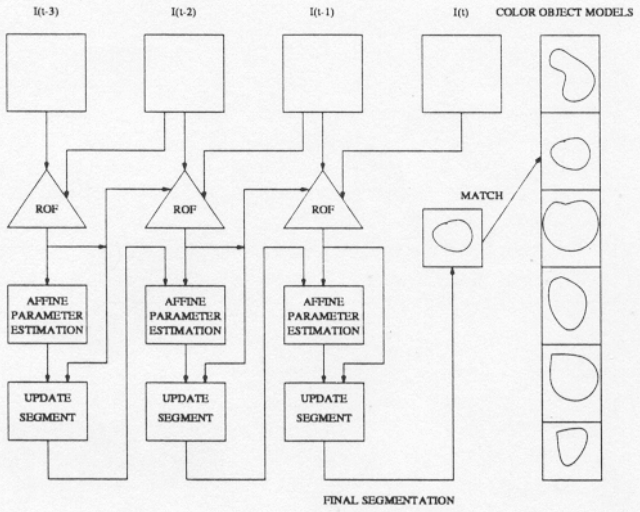


Figure 10: Incremental motion segmentation and object recognition using multi-resolution robust optical flow (ROF) estimation, affine parametric motion segmentation and color object recognition.

outliers

$$E_x = \sum_{x,y \in R} \rho_\sigma(u_a(x,y) - u(x,y)),$$
$$E_y = \sum_{x,y \in R} \rho_\sigma(v_a(x,y) - v(x,y)), \quad (5)$$

where R is the current region of support of the segmented object (initially equal to the full frame image size). u_a and v_a are horizontal and vertical affine motion flow vectors according to (3). (u, v) is the ROF estimated at the current instant, and $\rho_\sigma(x)$ is taken to be the Lorentzian robust estimator. We use successive over relaxation and graduated non-convexity regression methods [9, 22] to minimize this cost function by using a small number of iterations over a sequence of four image probes and updating the segmentation at every time instant.

The estimated affine motion parameters at the current time instant are then used to update the segmentation which serves as the region of support R for the next frame's affine minimization step. Fig. 11 shows the segmented background (showing two objects as outliers) and the segmentation of the outlier pixels into the object of interest (a blue fish).

At the end of the motion segmentation stage, the segmented objects are matched to color models using the color histogram intersection method. If the model object matches the peripheral segmented region, the animat localizes the recognized object using color histogram backprojection and foveates it to obtain a high-resolution view. It then engages in appropriate behavioral responses.

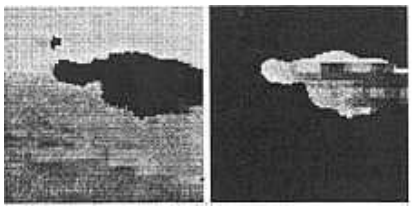
4.4 Behavioral Response to a Recognized Target

The behavioral center of the brain of the artificial animal assumes control after an object is recognized and fixated. If the object is classified as food the behavioral response would be to pursue the target in the fovea with maximum speed until the animat is close enough to open its mouth and eat the food. If the object is classified as a predator and the animat is a prey fish, then the behavioral response would be to turn in a direction opposite to that of the predator and swim with maximum speed. Alternatively, an object in the scene may serve as a visual frame of reference. When the animat recognizes a reference object (which may be another fish) in its visual periphery, it will fixate on it and track it in smooth pursuit at an intermediate speed. Thus, the fixation point acts as the origin of an object-centered reference frame allowing the animat to stabilize its visual world and explore its surroundings.

Fig. 12 shows a sequence of retinal images taken from the animat's left eye. The eyes are initially fixated on a red reference fish and thus the images are stabilized. In frames up to 285 a blue fish swims close by the animat's right side. The animat recognizes this as a reference fish and thus saccades the eyes to foveate the fish (frame 286 and beyond). It tracks the fish around, thereby exploring its environment. By foveating different reference objects, the animat can explore different parts of its world.

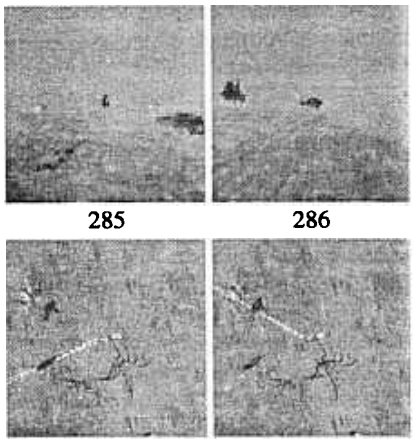
5 Summary and Conclusions

We have presented computer vision research carried out within the animat vision paradigm, which employs lifelike artificial fishes inhabiting a physics-based, virtual marine world. We have successfully implemented a set of active



Background Object

Figure 11: Results of incremental motion segmentation.



285 286

Figure 12: Retinal image sequence from the left eye of the predator (top) and overhead view (bottom) of the predator as it pursues a red reference fish (frames up to 285). A blue reference fish appears in the predator's right periphery and is recognized, fixated, and tracked (frames 286 and beyond). The white lines indicate the gaze direction.

vision algorithms for artificial fishes that integrate motion, stereo and color analysis. These algorithms support robust vision-guided navigation, visual perception, and obstacle recognition and avoidance abilities, enabling the animat to better understand and interact with its dynamic virtual environment. Our work should also be relevant to the design of active vision systems for physical robotics.

References

[1] H. E. Adler. *Fish Behavior: Why Fishes do What They Do*. T.F.H Publications, Neptune City, NJ, 1975.

[2] H. Arakawa and M. Etoh. An integration algorithm for stereo, motion and color in real-time applications. *IEICE Transactions on Information and Systems*, E78-D(12):1615 – 1620, December 1995.

[3] R. Bajcsy. Active perception. *Proceedings of the IEEE*, 76(8):996–1005, 1988.

[4] D. Ballard. Animate vision. *Artificial Intelligence*, 48:57–86, 1991.

[5] D.H. Ballard and L.E. Wixson. Object recognition using steerable filters at multiple scales. In *Proc. IEEE Workshop on Qualitative Vision*, pages 2 – 10, Los Alamitos, CA, June 1993.

[6] M.J. Black. Robust incremental optical flow. Technical Report YALEU/DCS/RR-923, Yale University, Dept. of Computer Science, 1992.

[7] M.J. Black and P. Anandan. A model for the detection of motion over time. In *Proc. Inter. Conf. Computer Vision (ICCV'90)*, pages 33–37, Osaka, Japan, 1990.

[8] M.J. Black and P. Anandan. A framework for the robust estimation of optical flow. In *Proc. Inter. Conf. Computer Vision (ICCV'93)*, pages 231–236, Berlin, 1993.

[9] A. Blake and A. Zisserman, editors. *Visual Reconstruction*. The MIT Press, Cambridge, Massachusetts, 1987.

[10] P.J. Burt, J.R. Bergen, R. Hingorani, R. Kolczynski, W.A. Lee, A. Leung, J. Lubin, and H. Shvaytser. Object tracking with a moving camera: An application of dynamic motion analysis. *Proc. IEEE Workshop Visual Motion*, pages 2 – 12, March 1989.

[11] M. Campani, A. Giachetti, and V. Torre. Optic flow and autonomous navigation. *Perception*, 24:253–267, 1995.

[12] C. Cedras and M. Shah. Motion-based recognition: A survey. *Image and Vision Computing*, 13(2):129–155, 1995.

[13] W.T. Freeman and E.H. Adelson. The design and use of steerable filters. *IEEE Trans. PAMI*, 13(9):891 – 906, September 1991.

[14] J.J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston, MA, 1979.

[15] W.E.L. Grimson, A.L. Ratan, G. Klanderman, and A. O'Donnell. An active visual attention system to play 'where's waldo'. In *Proc. of the Image Understanding Workshop*, volume 2, pages 1059 –1065, Monterey, CA, 13 - 16 Nov. 1994.

[16] B.K.P. Horn. *Robot Vision*. MIT Press, Cambridge, MA, 1986.

[17] C.L. Huang and Y.T. Chen. Motion estimation method using a 3D steerable filter. *Image and Vision Computing*, 13(1):21–32, 1995.

[18] M. Jagersand. Saliency maps and attention selection in scale and spatial coordinates: An information theoretic approach. In *Proc. Inter. Conf. Computer Vision (ICCV'95)*, pages 195–202, MIT, Cambridge, Massachusetts, June 20 - 23 1995.

[19] D.G. Jones and J. Malik. A computational framework for determining stereo correspondence from a set of linear spatial filters. In *Proc. Euro. Conf. Computer Vision (ECCV'92)*, pages 395 – 410, Portofino, Italy, 1992.

[20] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. *Proc. Image Understanding Workshop*, pages 121–130, 1981.

[21] D. Marr and T. Poggio. A computational theory of human stereo vision. *Proc. R. Soc. Lond. B.*, 204:301 – 328, 1979.

[22] T.F. Rabie and D. Terzopoulos. Motion and color analysis for animat perception. In *Proc. Thirteenth National Conf. on Artificial Intelligence (AAAI'96)*, pages 1090–1097, Portland, Oregon, August 4-8 1996.

[23] T.F. Rabie and D. Terzopoulos. Stereo and color analysis for dynamic obstacle avoidance. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'98)*, Santa Barbara, California, June 23-25 1998.

[24] T. Sanger. Stereo disparity computation using gabor filters. *Biological Cybernetics*, 59:405 – 418, 1988.

[25] E.P. Simoncelli and W.T. Freeman. The steerable pyramid: A flexible architecture for multi-scale derivative computation. In *IEEE Inter. Conf. on Image Processing*, pages 444 – 447, Washington, DC, October 23-26 1995.

[26] M. Swain and D. Ballard. Color indexing. *Inter. J. Computer Vision*, 7:11 – 32, 1991.

[27] M.J. Swain, R.E. Kahn, and D.H. Ballard. Low resolution cues for guiding saccadic eye movements. In *Proc. Inter. Conf. Computer Vision (ICCV'92)*, pages 737–740, 1992.

[28] D. Terzopoulos and T.F. Rabie. Animat vision: Active vision in artificial animals. In *Proc. Fifth Inter. Conf. Computer Vision (ICCV'95)*, pages 801 – 808, MIT, Cambridge, MA, June 20 - 23 1995.

[29] D. Terzopoulos and T.F. Rabie. Animat vision: Active vision in artificial animals. *Videre: Journal of Computer Vision Research*, 1(1):2–19, September 1997.

[30] D. Terzopoulos, X. Tu, and R. Grzeszczuk. Artificial fishes: Autonomous locomotion, perception, behavior, and learning in a simulated physical world. *Artificial Life*, 1(4):327–351, 1994.

[31] J.K. Tsotsos, S.M. Culhane, W. Wai, Y. Lai, N. Davis, and F. Nuflo. Modeling visual attention via selective tuning. *Artificial Intelligence*, 78(1-2):507–547, 1995.

[32] R.A. Young. Simulation of human retinal function with the Gaussian derivative model. In *Proc. Computer Vision and Pattern Recognition Conf. (CVPR'86)*, pages 564 – 569, 1986.